File Password

Introduction

Password-protected files are files that have been encrypted or secured with a password to restrict access. These files require the entry of the correct password to unlock and access their contents. Password protection is a common security measure used to safeguard sensitive or confidential information.

Here's why password-protected files are important and why we need them:

- **Confidentiality**: Password protection ensures the confidentiality of sensitive data. It prevents unauthorized users from viewing or accessing the contents of the file. This is especially crucial for files containing personal, financial, or proprietary information.
- **Data Security:** Password-protected files enhance data security. They provide an additional layer of protection beyond basic file access permissions. Even if someone gains access to the computer or storage device, they cannot open the file without the password.
- **File Sharing**: Password protection enables secure file sharing. Users can share password-protected files with trusted individuals while keeping the contents confidential from others.

How?

Encryption

Decryption

Protecting a file with a password involves encrypting its contents and securing the encryption key using the user's password. Here's a detailed explanation of the process:

Begin by selecting a robust encryption algorithm like Advanced Encryption Standard (**AES**). AES is widely recognized for its security.

A strong encryption key is essential. Use a cryptographically secure random number generator to create a long and **secure encryption key**. It is used to encrypt the actual file content.

Prompt the user to enter a password. This password will serve as the basis for deriving the **encryption key** using a Key Derivtion Function (KDF), such as PBKDF2 or bcrypt. Emphasize the importance of a strong, unique password. KDFs apply a **one-way transformation**, making it computationally infeasible to reverse and retrieve the original password.

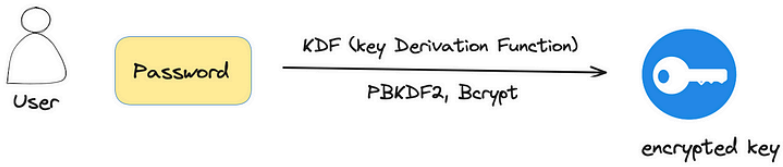

Fig 2.0: KDF generate encrypted key form user input password

Utilize the encryption key generated in step 4 to encrypt the file's contents using the chosen encryption algorithm (e.g., AES). This process produces ciphertext, which appears as random data and is not readable.

Safeguard the encrypted file, which includes the ciphertext and any associated metadata, in a secure location. This storage could be on disk, in a cloud service, or another secure medium.

Keep the encryption key (derived from the user's password) separate from the encrypted file. Securely store the key using methods like

When a user seeks to access the protected file, request their password.
Utilize the same KDF to derive the encryption key from the entered password.
Attempt to decrypt the file's contents using the derived key.

If the decryption process is successful (meaning the password is correct), the ciphertext transforms back into plaintext.
The user can then access the original content of the file.